



Article

Research on Autonomous Vehicle Path Planning Algorithm Based on Improved RRT* Algorithm and Artificial Potential Field Method

Xiang Li , Gang Li * and Zijian Bian

School of Automobile and Traffic Engineering, Liaoning University of Technology, Jinzhou 121001, China; 221285015@stu.lnut.edu.cn (X.L.); bianzijian295@163.com (Z.B.)

* Correspondence: qcxyligang@lnut.edu.cn

Abstract: For the RRT* algorithm, there are problems such as greater randomness, longer time consumption, more redundant nodes, and inability to perform local obstacle avoidance when encountering unknown obstacles in the path planning process of autonomous vehicles. And the artificial potential field method (APF) applied to autonomous vehicles is prone to problems such as local optimality, unreachable targets, and inapplicability to global scenarios. A fusion algorithm combining the improved RRT* algorithm and the improved artificial potential field method is proposed. First of all, for the RRT* algorithm, the concept of the artificial potential field and probability sampling optimization strategy are introduced, and the adaptive step size is designed according to the road curvature. The path post-processing of the planned global path is carried out to reduce the redundant nodes of the generated path, enhance the purpose of sampling, solve the problem where oscillation may occur when expanding near the target point, reduce the randomness of RRT* node sampling, and improve the efficiency of path generation. Secondly, for the artificial potential field method, by designing obstacle avoidance constraints, adding a road boundary repulsion potential field, and optimizing the repulsion function and safety ellipse, the problem of unreachable targets can be solved, unnecessary steering in the path can be reduced, and the safety of the planned path can be improved. In the face of U-shaped obstacles, virtual gravity points are generated to solve the local minimum problem and improve the passing performance of the obstacles. Finally, the fusion algorithm, which combines the improved RRT* algorithm and the improved artificial potential field method, is designed. The former first plans the global path, extracts the path node as the temporary target point of the latter, guides the vehicle to drive, and avoids local obstacles through the improved artificial potential field method when encountered with unknown obstacles, and then smooths the path planned by the fusion algorithm, making the path satisfy the vehicle kinematic constraints. The simulation results in the different road scenes show that the method proposed in this paper can quickly plan a smooth path that is more stable, more accurate, and suitable for vehicle driving.

Keywords: autonomous vehicle; RRT* algorithm; path planning; fusion algorithm; curvature; artificial potential field method



Citation: Li, X.; Li, G.; Bian, Z. Research on Autonomous Vehicle Path Planning Algorithm Based on Improved RRT* Algorithm and Artificial Potential Field Method. *Sensors* **2024**, *24*, 3899. <https://doi.org/10.3390/s24123899>

Academic Editors: Jun Chen and Wen-Chiao Lin

Received: 15 May 2024

Revised: 13 June 2024

Accepted: 13 June 2024

Published: 16 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The casualties and economic losses caused by road traffic accidents are increasing year by year. Every year, there are traffic accidents caused by improper driving behavior and other related reasons [1]. Driverless cars can well reduce road congestion and greatly reduce traffic accidents caused by human errors [2].

One of the crucial technologies for self-driving vehicles is path planning [3]. It plans the safe driving path according to the map information, sensor data, and target location information and controls the steering and speed of the vehicle to ensure that the vehicle runs safely and efficiently to the destination [4]. The commonly used path-planning algorithms are the intelligent method, the random sampling method, the curve interpolation

method, and so on. It can also be classified by global path planning and local path planning. The global path planning algorithm has two methods: random sampling and graph-based search, such as the Dijkstra algorithm [5], the A* algorithm [6], PRM algorithm [7], and the RRT* algorithm [8]. Among them, the RRT* algorithm finds extensive application in the realm of mobile robots and unmanned vehicles, and many researchers have improved the RRT* algorithm. The RRT* algorithm proposed by Karaman et al. [9] introduces the functions of a reselection parent node and reconnection to optimize the search results. However, its search efficiency is suboptimal. Cong et al. [10] proposed an RRT* algorithm based on a hybrid sampling strategy, which effectively reduced the sampling randomness of the algorithm. The HBAI-RRT* algorithm proposed by Lin et al. [11] in their study combines the greedy heuristic method with an adaptive adjustment strategy to reduce the sampling area and improve search efficiency. Local path planning algorithms more frequently employ the artificial potential field (APF) method, which can lead to local minima and goal unreachability issues during the planning process [12]. After conducting an in-depth study, Li et al. [13] adopted the concept of the artificial potential field and improved strategies such as adding distance adjustment factors to address the limitations of the traditional artificial potential field method. Xu [14] engineered a spring mechanism in the repulsive velocity potential field to effectively eliminate the flutter phenomenon near the obstacle. Zhai [15] and others added an adjustment factor and a judgement coefficient to the potential field to optimize the potential field function and improve driving comfort and the safety of the planned path. Many other researchers have proposed fusion algorithms combining the advantages of the two algorithms. Huang [16] introduced a fusion algorithm that merges the RRT algorithm with the artificial potential field method. The expansion of the RRT random tree incorporates probability values and a gravity component. With this method, simulation results demonstrate notable improvements in time efficiency, path length, and iteration count. Zhang [17] introduced a novel path-planning algorithm that enhances obstacle avoidance speed and quality by combining the A* algorithm with the artificial potential field method. This fusion also effectively addresses the inefficiencies of the A* algorithm when dealing with complex scenes. Wu [18] proposed a path planning algorithm that integrates the artificial potential field method with the particle swarm optimization algorithm. This approach allows for the real-time generation of obstacle avoidance paths and significantly improves the vehicle's stability when avoiding obstacles. Dasiah et al. [19] proposed an improved RRT* algorithm that realizes directional fast search based on sampling angle constraints, which can effectively find better paths in complex environments and significantly improve the convergence rate of the algorithm. In terms of optimality and optimization in path planning problems, Nguyen et al. [20] proposed a research method that integrates formation control and optimal control, taking into account the kinematic and dynamic models of each vehicle. Experiments showed the effectiveness of the control strategy. Shi et al. [21] studied the distributed time-varying output formation tracking problem of heterogeneous multi-agent systems with different dimensions and parameters. They designed an optimal tracking controller by adopting model-free reinforcement learning technology and designing the compensation input for each follower. The simulation results show that the output formation tracking error eventually approaches zero.

Although the above scholars have improved the path planning algorithm, there are still problems that cannot deal with the unknown obstacles emerging in the environment and cannot be applied to structured roads. Although some scholars consider real-time obstacle avoidance, the vehicle collision range is not taken into account, and the final path is not optimized. After consulting the literature, there is little research on the path planning algorithm in the curved structured road scene, and there are some problems, such as low search efficiency. Compared with the optimization of the path tracking control in study [20] and study [21], this paper mainly studies the efficiency and quality of the planned path of an unmanned vehicle, which provides a high-quality path for tracking control and improves the efficiency and stability of tracking. Therefore, this paper designs

a method of vehicle driving guided by global path nodes and local planning to avoid obstacles. In order to solve the above path planning problems, this is necessary. The RRT* global path planning algorithm with more sampling nodes is adopted to ensure that more nodes play a guiding role, and the APF local path planning algorithm, which is better at considering the collision range of obstacles and structured road constraints, is adopted to solve the path planning problem of unknown obstacles on structured curved roads. This paper proposes a fusion algorithm that combines the Improved RRT* algorithm and the Improved Artificial Potential Field method. The objective is to tackle the path planning challenge that arises when encountering a prominently unknown obstacle on a structured road. By introducing the concepts of gravitational and repulsive fields into RRT* node sampling, adding probabilistic sampling optimization, and enhancing the purposefulness of node sampling, we improve the RRT* algorithm. Considering the curvature of the road and designing the adaptive step size, which makes the sampling more efficient and time-consuming and solves the problem of possible oscillation when expanding near the target point, subsequently, the nodes of the planned path undergo branching and constraints to finalize the path post-processing phase, enhancing the overall path quality. Enhancements to the artificial potential field method include the design of obstacles, optimization of obstacle avoidance constraints, addition of a road boundary repulsive potential field and optimization of the repulsive function, and the introduction of virtual gravitational points. The goal unreachability problem is solved, and the local optimum is reduced to improve the quality of locally planned paths. Finally, the fusion algorithm significantly improves the performance of obstacle avoidance on structured roads. The planned roads are smoothed to ensure that they are suitable for vehicles.

2. Traditional Algorithm

2.1. RRT* Algorithm

The RRT algorithm conducts random sampling in unoccupied space and continuously expands random tree branches until a branch contains or approaches the target point, or until the algorithm reaches its iteration limit, at which point the planning process concludes [22].

From reference [9], it is known that the RRT* algorithm has the following theorems:

Theorem 1. (*asymptotic optimality of the RRT* algorithm*): when the number of iterations tends to infinity, the path quality of the RRT* algorithm tends to be optimal; that is, the algorithm has asymptotic optimality.

Theorem 2. (*probabilistic completeness of the RRT* algorithm*): if there is a feasible path from the starting point to the end point, the RRT algorithm will almost certainly find a feasible path with an infinite number of iterations.

In the RRT algorithm, the root node of the random tree is the starting point x_{start} , and a random point x_{rand} is generated in the blank area of the map. Then, the whole random tree is searched, and $x_{nearest}$ is selected as the nearest node to x_{rand} . Connect x_{rand} and $x_{nearest}$, and expand a new node on the connected straight line at a set step size of x_{new} . Next, we subject the new node to collision detection. If it does not collide with an obstacle, we add it to the randomly expanded tree. If a collision occurs, then we delete the new node. Continue the described steps until either the target node is appended to the random tree node or the search concludes upon reaching the specified number of iterations. The planned path is derived by retracing from the target point to the starting point. The random sampling process is illustrated in Figure 1.

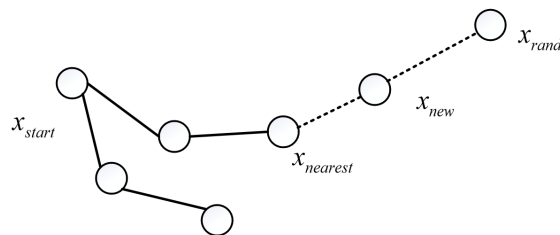


Figure 1. The node extension process.

The RRT* algorithm enhances the RRT algorithm by incorporating improved strategies for parent node selection and reconnecting when sampling new nodes [23], which reduces the generated redundant paths and makes the planned paths better. The procedure for reselecting parent nodes is depicted in Figure 2.

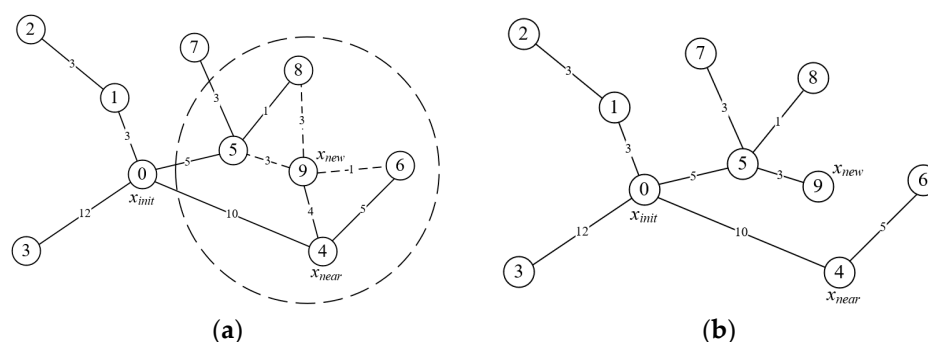


Figure 2. The process of reselecting the parent node: (a) the constrained search range when reselecting parent nodes; (b) the node situation after reselecting the parent node.

With node x_{new} as the center of the circle, form a constrained search range within the defined radius, as shown in Figure 2a. The neighboring nodes in it are used as candidates for the parent node of x_{new} . When solving for taking each candidate as a parent node, the cost from the starting point to x_{new} is calculated and compared with the cost of the original path; the candidate node with the lowest cost is chosen as the new parent node. This results in the least costly path within the constrained radius size at this point in time, and the candidate node for the path at this point in time is selected in place of the original node [24]. The path cost with the candidate point as the parent node is shown in Table 1, and the parent node should be updated to node 5, as shown in Figure 2b.

Table 1. Constrained range path cost.

Candidate Point	Trails	Path Distance
5	0-5-9	0-5-9 = 8
6	0-4-6-9	0-4-6-9 = 16
8	0-5-8-9	0-5-8-9 = 9

After completing the parent node reselection, a reconnection process is also required to prune the random tree [25], thereby further reducing the path length. Within the same constraints, as shown in Figure 3a, find all the neighboring nodes of node x_{new} with x_{new} as the parent of each neighboring node and find the cost of each path, i.e., if the cost is less than the cost of the path of the original parent of the neighboring node, update its parent to node x_{new} . The reconnection process is shown in Figure 3. From Table 2, the parent node of node 6 should be updated to node 9, as shown in Figure 3b.

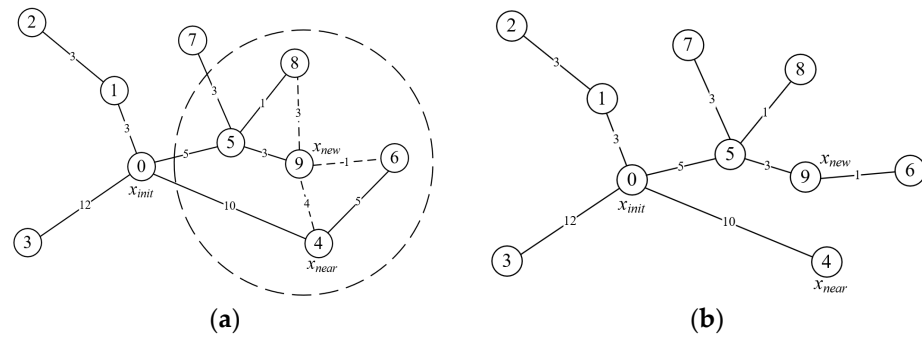


Figure 3. Reconnection process: (a) the constrained search range for reconnection; (b) the post-node situation after reconnection.

Table 2. Reconnection process path cost.

Neighboring Node	Original Consideration	New Pathway	New Price
4	10	0-5-9-4	0-5-9-4 = 12
6	10 + 5 = 15	0-5-9-6	0-5-9-6 = 9
8	3 + 5 + 1 = 6	0-5-9-8	0-5-9-8 = 11

Parent node reselection and reconnection strategies complement and interact with each other to significantly improve the path quality.

Although the parent node reselection and reconnection strategy in RRT* can improve the path quality, it increases the time cost and still has a slow convergence speed, which requires a large number of iterations to approach the better path. When dealing with a complex obstacle environment, especially in the case of a narrow or curved channel, there are some problems, such as unstable path quality, low algorithm efficiency, even planning failure, redundant nodes, and an unsmooth path, and there is still a lack of targeted sampling. Therefore, the improved strategy should be introduced into the RRT* algorithm according to Lemma 1.

Lemma 1. (improving the efficiency of the RRT* algorithm): the improved RRT* algorithm reduces the number of invalid sampling points by introducing heuristic information or an optimization strategy, thus improving the efficiency of the algorithm.

2.2. Traditional Artificial Potential Field Method

The artificial potential field method views the vehicle motion space as a virtual force field space, with the target point on the vehicle producing gravitational force and the obstacle producing repulsive force. Therefore, the vehicle moves under the effect of these two forces.

Within the virtual realm of a force field, the attraction of a target point to a vehicle is directly proportional to the distance between them. The equation for the gravitational potential field function is provided below.

$$U_{att}(d) = \frac{1}{2}k_{att}d^2(p, p_g) \quad (1)$$

where k_{att} is the gravitational potential field gain coefficient, p is the current location of the vehicle, p_g denotes the position of the target point, and $d(p, p_g)$ is a vector of magnitude representing the distance between the vehicle and the target point in the direction of the vehicle pointing to the target point. The gravitational force function is obtained from the above equation, as shown in the following equation.

$$F_{att}(d) = k_{att}d(p, p_g) \quad (2)$$

When the vehicle is out of its repulsive potential field's range, the obstacle exerts no repulsive force. Only within the range of the repulsive potential field will the obstacle's repulsive force affect the vehicle; the greater the distance between the two, the smaller the vehicle's exposure to the repulsive potential energy value, and vice versa. The following equation illustrates the repulsive potential field function.

$$U_{req}(d) = \begin{cases} \frac{1}{2}k_{rep}\left(\frac{1}{d(p,p_0)} - \frac{1}{d_0}\right)^2 \frac{1}{d^2(p,p_0)}, & 0 \leq d(p, p_0) \leq d_0 \\ 0, & d(p, p_0) > d_0 \end{cases} \quad (3)$$

where k_{rep} is the repulsive potential field gain coefficient, p is the current position of the vehicle, p_g denotes the position of the target point, and $d(p, p_g)$ is a vector of magnitude representing the Euclidean distance between the vehicle and the obstacle in the direction of the vehicle pointing towards the obstacle. The repulsive force function on the vehicle can be calculated from the above equation as follows:

$$F_{req}(d) = \begin{cases} k_{rep}\left(\frac{1}{d(p,p_0)} - \frac{1}{d_0}\right), & 0 \leq d(p, p_0) \leq d_0 \\ 0, & d(p, p_0) > d_0 \end{cases} \quad (4)$$

In the process of driving, the vehicle will be affected by both the gravitational field and the repulsive field. The combined potential field to which it is subjected is as follows:

$$U(d) = U_{att}(d) + \sum_{i=0}^n U_{rep}(d) \quad (5)$$

where n is the number of obstacles that have a repulsive effect on the vehicle.

From the combined potential field, we can obtain the combined force applied to the vehicle during its motion, and the combined force expression is given in the following equation:

$$F(d) = F_{att}(d) + \sum_{i=0}^n F_{rep}(d) \quad (6)$$

However, the traditional artificial potential field method is prone to local minimum problems and unreachable targets. As a result, the unmanned car may fall into a "trap area" (where the resultant force on the vehicle is zero). For example, in the face of U-shaped obstacles, it is easy to fall into the local minimum area, so that the target point cannot be reached. It is also possible that the unmanned vehicle cannot reach the target position because the strong repulsive force will produce concussive behavior, which will eventually lead to the failure of path planning.

3. Improved RRT* Algorithm

3.1. Introducing the Concept of Artificial Potential Fields

3.1.1. Introducing the Gravitational Component

In the RRT* algorithm, the gravitation between the random node and the target point is increased to guide the random tree to grow to the target point more rapidly and reduce the randomness. The schematic diagram is shown in Figure 4. In order to change the resultant force of the growth direction, the gravity function $G(n)$ is added to each new node n as follows:

$$F_1(n) = Rd(n) + G(n) \quad (7)$$

where $F_1(n)$ is the function of the new node, $Rd(n)$ is the random function of new nodes, and $G(n)$ is the objective gravity function.

The gravity function $G(n)$ is as follows:

$$G(n) = \rho \times g \times \frac{x_{goal} - x_{near}}{|x_{goal} - x_{near}|} \quad (8)$$

where ρ is the step size, g is the gravity gain factor, x_{goal} is the target position vector, $|x_{goal} - x_{near}|$ is the magnitude of the geometric distance between node x_{goal} and node x_{near} .

Random function $Rd(n)$:

$$Rd(n) = \rho \times \frac{x_{goal} - x_{near}}{|x_{goal} - x_{near}|} \quad (9)$$

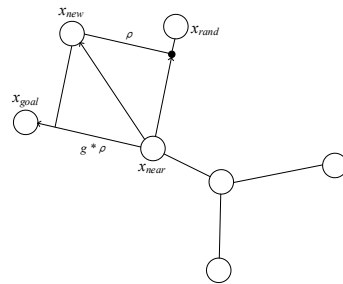


Figure 4. Random tree growth with increasing gravitational components.

3.1.2. Introduction of Repulsive Force Component

When the RRT* algorithm expands the tree, it makes the random tree have a certain distance from the obstacle by increasing the repulsive component, as shown in Figure 5. The repulsive force component is added to the new nodes generated in the process of random tree expansion. Add the repulsion function $T(n)$ at node n , as follows:

$$F_2(n) = Rd(n) + T(n) \quad (10)$$

where $F_2(n)$ is the new node in the search process, $Rd(n)$ is the random growth function, and $T(n)$ is the repulsion function.

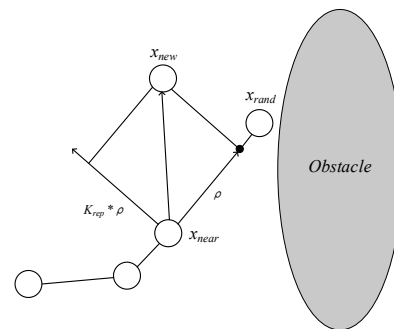


Figure 5. Random tree growth with increasing repulsion component.

The obstacle repulsion function $T(n)$ is as follows:

$$T(n) = \begin{cases} 0, & p(x) > p_0 \\ \rho k_{rep} \left(\frac{1}{p(x)} - \frac{1}{p_0} \right) \frac{1}{p(x)^2} \frac{\partial(x_{near} - x_{obstacle})}{\partial x_{near}}, & p(x) \leq p_0 \end{cases} \quad (11)$$

where k_{rep} is the repulsive force gain coefficient, $p(x)$ is the shortest distance from x_{rand} to the obstacle, p_0 is the distance affected by the obstacle on the node, and $x_{obstacle}$ is the position vector of the obstacle.

The new node generating function $F(n)$ is as follows:

$$F(n) = F_1(n) + F_2(n) \quad (12)$$

where $F_1(n)$ represents the new node function of the random tree in the gravitational field, and $F_2(n)$ represents the new node function of the random tree in the repulsion field. It is known that $F(n)$ is not only determined by x_{rand} , which avoids random sampling of the RRT*.

3.2. Probabilistic Sampling Optimization

Because of the strong randomness in sampling, the RRT* algorithm often causes random trees to deviate from growth, which seriously affects the efficiency of the algorithm. It can be seen from Theorem 2 that in order to find a feasible path faster, the random tree of the algorithm can grow more purposefully, according to Lemma 1. Therefore, probabilistic sampling optimization is designed to improve target orientation and reduce redundant searches [26].

Firstly, the obstacle-free area of the target point is determined, that is, the minimum distance from target x_{goal} to all obstacles, which is called the convergence radius of target point R_g . As depicted in Figure 6, a region circle is generated with R_g as the radius and x_{goal} as the center, and the inner circle is the convergence region of the target point. Its characteristic is that when the random tree grows within this range, it can direct itself to the end point.

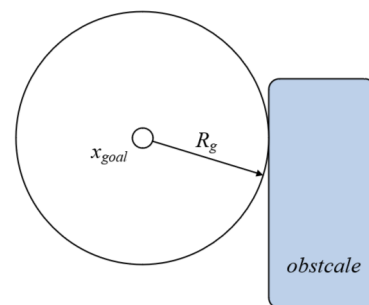


Figure 6. Schematic diagram of convergence region of target point.

In the strategy of target bias, a target bias probability value p_{target} is randomly generated by using the random number generating function, and then a random probability value p is obtained, which is greater than 0 and less than 1. When $p \geq p_{target}$ is obtained, then the sampling point is randomly generated. If $p < p_{target}$ is in the convergence region of the target point, the random extension tree grows near the target point with a certain probability by randomly generating a point as a sampling point. This method can accelerate the convergence rate to the target node. The sampling point-generating function is as follows:

$$X_{rand} = \begin{cases} GoalArea(), & p < p_{target} \\ Sample(), & p \geq p_{target} \end{cases} \quad (13)$$

where the *GoalArea* function represents the random generation of a sampling point in the convergence region of the target point, the *sample* function represents the random generation of a sampling point in the global range, and x_{rand} represents the randomly generated sampling point.

The probabilistic sampling optimization of the global path search is carried out [27]. When $p \leq p_{target}$, the target offset sampling is adopted, that is, a point is randomly generated in the target convergence domain as the sampling point. When $p_{target} < p \leq p_{goal}$, there is uniform random sampling in the search space.

When a new node coordinate x_{new} is generated, it is determined whether the x_{new} is in the target point convergence region. If so, the x_{new} is directly connected to the terminal x_{goal} , but the step size ρ may be set to be larger than the target convergence radius R_g , causing the random tree to oscillate when it expands near the x_{goal} . This will cause the random tree to have limitations at the target point, as depicted in Figure 7.

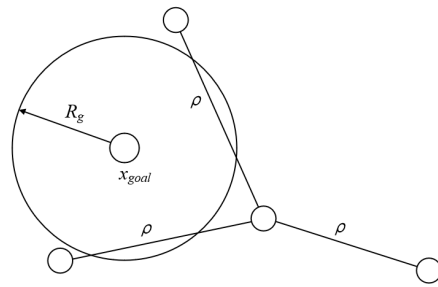


Figure 7. The limitation of the random tree near the target point.

In order to avoid this situation and make it get the optimal path faster, when $p \geq p_{goal}$ occurs, all of the random tree nodes are traversed to find the node x_k with the lowest cost of c_{min} , as shown in the following formula:

$$c_{min} = C^k + d(x_k, x_{goal}) \quad (14)$$

where C^k is the actual path cost from x_{init} to x_k , and $d(x_k, x_{goal})$ is the Euclidean distance from node x_k to target point x_{goal} . Connect x_k and x_{goal} , then perform collision detection on the path from x_k to x_{goal} . If no collision occurs, x_k is directly used as the parent node of x_{goal} , and then the path is generated. If a collision occurs, it is changed to uniform random sampling, where p_{target} and p_{goal} are between 0 and 1.

In the initial target offset strategy, the target bias probability of the RRT* algorithm is fixed, and its adaptability to the scene is poor. In order to improve adaptability, a probability adaptive target bias strategy is designed to expand the tree growth combined with the gravity component constraints mentioned above to shorten the planning time [28].

Before generating a new sampling point, a decision circle is generated with the center of x_{new} as the center and the connection between x_{near} and x_{new} as the radius R , as shown in Figure 8.

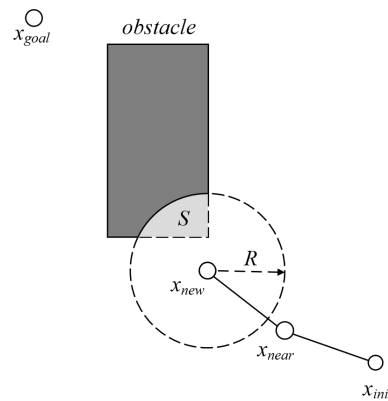


Figure 8. Construct the decision area.

In Figure 8, x_{new} is the center of the circle, R is the radius, and S is the area of the obstacle in the circle. The proportional value P can be obtained from the area value S , as shown in the following formula:

$$P = \frac{S}{(\pi * R)^2} \quad (15)$$

Then change the scale value P to the target offset probability value p_{target} ; the formula is as follows:

$$p_{target} = 1 - e^{-(P*4.5)} \quad (16)$$

In the process of determining circle generation, the proportional value P will change with the change of radius R and occupied area value S , while the probability value p_{target}

will change with the change of proportional value P , so as to realize the self-adaptation of target bias probability. After the target bias probability value p_{target} is obtained and compared with the random probability value p , sample according to the probability sampling optimization method mentioned above.

When the target offset sampling is adopted, the expansion tree grows to the target point, and the random node x_{rand} will appear in the direction of the x_{goal} and x_{near} connections. Set the step size to ρ , and generate the formula for the new node x_{new} as follows:

$$x_{new} = x_{near} + \frac{x_{rand} - x_{near}}{\|x_{rand} - x_{near}\|} * \rho \quad (17)$$

When uniformly randomly sampled in the search space, the extension tree grows in the direction of the resultant force of random and gravitational components, so that the expansion of the new node will not deviate from the shortest path and enhance the goal bias of path sampling. The diagram of the new node is shown in Figure 4 above. The following formula will generate the new node x_{new} :

$$x_{new} = x_{near} + \left(\frac{x_{rand} - x_{near}}{\|x_{rand} - x_{near}\|} * \rho + \frac{x_{goal} - x_{near}}{\|x_{goal} - x_{near}\|} * g * \rho \right) \quad (18)$$

3.3. Adaptive Step Size Strategy

In the process of global planning sampling, it can be seen from Theorem 1 that the algorithm should speed up the search speed, make its iterative progress faster, and improve the path quality. Therefore, in this improvement, the adaptive step size should be designed according to Lemma 1, so as to accelerate the sampling efficiency and improve the path quality. Different road environments need different steps. If the step size is too short, when there are few obstacles near the vehicle, or if the curvature of the road is small, the search takes a long time and reduces the search efficiency. If the step size is too large, there are many obstacles in the environment, or if the road has a large curvature, it may fall into a stagnant state and be unable to generate a feasible path. In order to further optimize the step size selection, an adaptive step size strategy is proposed. The step size ρ uses the number of obstacles near the vehicle and the curvature of the road so as to improve the adaptability of the vehicle to the environment. The formula for calculating curvature k is as follows:

$$k = \frac{\ddot{y}}{(1 + (\dot{y})^2)^{\frac{3}{2}}} \quad (19)$$

where y is the road function. As the vehicle runs, there is continuous acquisition of vehicle location information and lane conditions, and the curvature of the lane is calculated.

The adaptive step size policy formula is as follows:

$$\rho = \rho_0 e^{\lambda(p-k-q)} \quad (20)$$

where ρ_0 is the initial step, p is the threshold of the number of obstacles, k is the curvature of the road, q is the number of obstacles, and λ is a value between 0 and 1.

When the number of obstacles around vehicle q and the road curvature k are zero, more than double the initial step size of ρ_0 is selected to generate new nodes. When the number of obstacles q is less than or equal to the threshold p and the curvature of the road is small, it means that the vehicle is driving on a road similar to a straight line and the number of obstacles around is small, and a new node is created by using a step close to the initial step size of ρ_0 . When the number of obstacles q is greater than the threshold p , it shows that there are a large number of obstacles around the vehicle. When there is a large curvature on the road, it means that the road environment where the vehicle is located is a large curve. Consider that the vehicle cannot come into contact with obstacles; a new node is created

by 0–1 times the initial step length of ρ_0 . After the above process, the sampling step size can automatically change the applicable value in different road environments, improving the adaptability of the algorithm and enhancing the robustness of path planning.

When the target point is on the straight road and there are few obstacles nearby, a longer step size should be selected, but using a larger expansion step size will produce the oscillation phenomenon shown in Figure 7, resulting in an increase in path planning time and redundant nodes. In the optimization of probabilistic sampling, although it is possible to take x_{goal} as the parent node near the target point, there is still probability. Therefore, it is necessary to further optimize the algorithm's performance. The distance $Distance$ between $x_{nearest}$ and x_{goal} is calculated before each expansion. If $Distance > \rho$, the expansion step size remains the same; otherwise, the expansion step size is adjusted to the $Distance$ more suitable for growing near x_{goal} , and the oscillation problem is solved.

3.4. Global Path Post-Processing

After the above-mentioned improvement strategies, the algorithm in this paper can quickly plan the initial path, but there may still be some tortuous paths and a large number of unnecessary nodes [29]. First of all, for the redundant paths, the triangle principle is used to design the optimization node strategy, and then for some of the tortuous paths, the constraint based on the minimum turning radius is established [30].

3.4.1. Node Pruning Strategy

The principle of the node pruning strategy is depicted in Figure 9. The dashed line represents the initial path planned by the improved RRT* algorithm without incorporating a node pruning strategy. The planning path is formed by the nodes in the initial node set $\{Q_1, Q_2, Q_3, Q_4, Q_5, Q_6\}$, where Q_1 and Q_6 are the starting point and target point, respectively. After introducing the node pruning strategy, take the target point Q_6 as the root node, and each node in the initial node set is connected successively based on Q_6 , and collision detection is carried out. It is found that line segment Q_6Q_1 , line segment Q_6Q_2 , and Q_6Q_3 will collide with obstacles, but there is no collision when Q_6 is directly connected to Q_4 , so Q_4 is added to the set of optimized nodes. Then, taking Q_4 as the starting point of collision detection, Q_4 can only connect with Q_3 without collision, so Q_3 is added to the set of optimized nodes. Connect each node in the original path in turn and carry out a new round of collision detection. We can know that when Q_1 can be directly connected to Q_3 , when the starting point is added to the optimized node set, it indicates that the node reconnection ends, that is, Q_1 is put into the optimized node set. According to the order of the nodes in the optimized node set, the optimization path is obtained. The path is depicted by the solid line in Figure 9. Compared with the original path, it is observed that the optimized path features a substantial reduction in node count and effectively decreases the path's length cost.

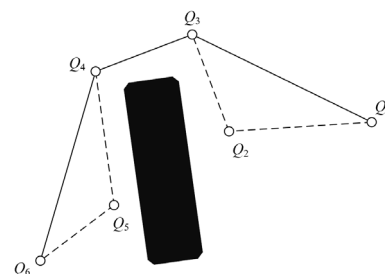


Figure 9. Schematic diagram of node branch pruning strategy.

3.4.2. Node Optimization

After the path is treated by the node branch and shear, the joint angle of part of the path cannot meet the vehicle dynamics constraint, so it is necessary to optimize the route with too large a bend angle to ensure that the turning angle at the turning point of the

path satisfies the vehicle kinematic constraints [31]. And the continuous folding angle is optimized to make the turning angle more uniform and stable. The process is illustrated in Figure 10.

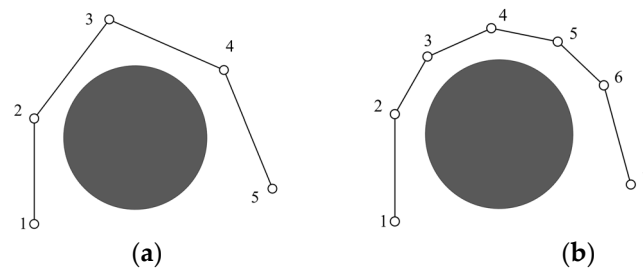


Figure 10. Schematic diagram of node optimization strategy: (a) before node optimization; (b) after node optimization.

As shown in Figure 10a, in the planned path, the angle between node 2 and node 3 and node 4 and node 5 does not conform to the vehicle kinematic constraint, so it needs to be optimized. This problem is solved by adding auxiliary nodes, such as nodes 3, 4, and 5 in Figure 10b, which flatten the initial angle so as to meet the minimum turning radius constraint of the vehicle.

4. Improved Artificial Potential Field Method

4.1. Obstacle Avoidance Constraint

When modelling obstacles in the road environment, the threat in the longitudinal direction is generally greater than that in the horizontal direction, so the longitudinal distance between the main vehicle and the environmental obstacles should be fully considered when establishing obstacle avoidance constraints. Therefore, the surrounding obstacles are modelled as ovals. The major axis of the ellipse corresponds to the longitudinal axis of the obstacle, and the short axis of the ellipse corresponds to the horizontal axis of the obstacle. Considering that the obstacle has speed, this will impact the main vehicle's obstacle avoidance performance. Therefore, a dynamic ellipse model of environmental obstacles, which can change with speed, is designed. The long axis length of the ellipse is associated with the velocity parameter. Thus, velocity is directly related to the length of the dynamic ellipse's major axis, indicating that a larger influence area corresponds to a lower collision probability [32]. Environmental obstacles are depicted, as illustrated in Figure 11.

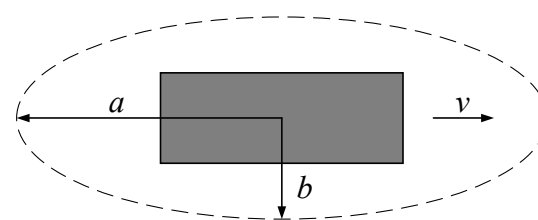


Figure 11. Schematic diagram of environmental obstacles.

The major and minor axes of the dynamic ellipse are denoted as follows:

$$a = \frac{L}{2} + k_a + k_r v \quad (21)$$

$$b = \frac{W}{2} + k_b \quad (22)$$

where L represents the length of the obstacle, W represents the width of the obstacle, v denotes the speed of the obstacle, k_a and k_b are horizontal and vertical security redundancy distances, and k_r is the safety redundancy time.

In the above modelling, the collision can be judged by judging whether there is an overlap between the main vehicle and the ellipse boundary. The boundary of the ellipse is divided into many circles, and several circles are made on the boundary line of the ellipse with a width of 1/2 of the vehicle as the radius, thus forming a new ellipse, that is, the extended ellipse. It is equivalent to increasing the collision range by half a vehicle's width on the basis of the original obstacle ellipse. The distance between the vehicle and the obstacle is increased, and the path safety is improved. The extended ellipse is shown in Figure 12.

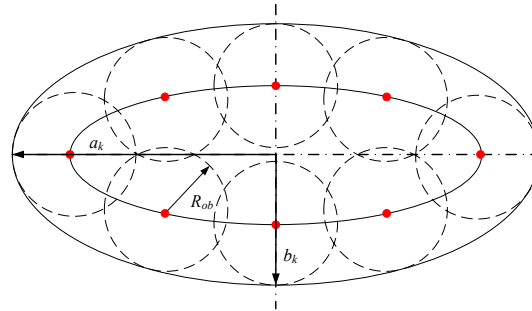


Figure 12. Extended ellipse.

The major and minor axes of an extended ellipse are expressed as follows:

$$a_k = \frac{L}{2} + k_a + k_r v + R_{ob} \quad (23)$$

$$b_k = \frac{W}{2} + k_b + R_{ob} \quad (24)$$

When the vehicle is driving on the planned path, the point mass model is used to simplify the vehicle, and collision detection is performed by assessing the spatial relationship between the point mass model and the extended ellipse. If the center of mass lies within or on the extended ellipse, a collision is deemed to have occurred, and the collision constraint is not met. Conversely, if the center of mass is located outside the extended ellipse, no collision is considered to have taken place, and the collision constraint is satisfied.

4.2. Road Repulsion Potential Field

During main car operation on a two-lane road, the lane line and lane boundary should cause repulsion to the moving vehicle, and when there are no obstacles in the road, the vehicle should not leave the current lane unless it is necessary to change lanes when obstacles are encountered. Vehicles are not allowed to press the line or cross the road boundary to avoid traffic accidents.

In the road repulsion potential field model design, ensuring that the repulsion potential field at the lane boundary is the highest serves to confine the vehicle within the lane boundary [33]. While preserving the boundary repulsion potential field, the lane repulsion potential field is incorporated. To ensure smooth lane changes for the main vehicle, the maximum value of the lane's repulsive potential field must be lower than that of the obstacle. A novel road repulsion potential field is formulated to prevent vehicle lane changes in the absence of obstacles. The repulsion potential field for a vehicle travelling along the centerline of the current lane is minimized to constrain the vehicle's ability to remain in the lane's centerline. The road repulsion potential field U_{road} is obtained by superposing the lane line potential field U_l and the road boundary potential field U_r . The formula for the road repulsion potential field U_{road} is expressed as follows:

$$U_{road} = U_l + U_r \quad (25)$$

The lane line potential field U_l can make the vehicle keep a certain distance from the lane and make it move towards the center of the lane. It should be guaranteed to overcome

the potential field when encountering obstacles, which requires that the lane potential field be small enough to ensure driving safety. The lane line potential field established in this paper is expressed by the following formula:

$$U_l = \sum_i^{N_{ls}-1} U_{l,i} \quad (26)$$

$$U_{l,i} = A_l \exp\left(-\frac{(y - y_{c,i})^2}{2\sigma^2}\right) \quad (27)$$

where A_l represents the intensity coefficient of the lane potential field, $y_{c,i}$ is the central position of the Gaussian function, the i lane coordinate is generally set on the centerline of the lane, and σ determines the range of the potential field, that is, the sensitivity of the vehicle to the lane centerline.

Set the road boundary potential field U_r to infinity to limit vehicle crossing, as shown in the U_r expression:

$$U_r = \sum_j^N U_{r,j} \quad (28)$$

$$U_{r,j} = \frac{1}{2}\eta\left(\frac{1}{y - y_{0,j}}\right)^2 \quad (29)$$

where η is the potential field factor of the boundary road, $y_{0,j}$ is the y coordinate of the j road boundary, and N represents the number of road boundaries. Figure 13 shows the road potential field diagram of the Dow repulsion.

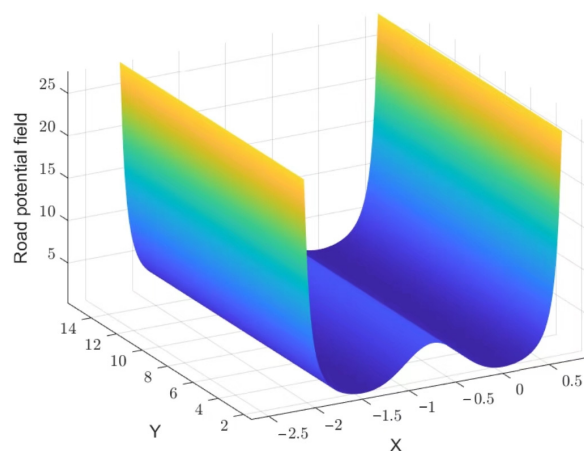


Figure 13. Road repulsion potential field diagram.

4.3. Increase Distance Factor

To address the issue that the target point is unreachable, it is necessary to optimize the repulsion function and add the distance factor d_g^2 [34]. The improved repulsive potential field function is as follows:

$$U_{rep}(d, \alpha) = \begin{cases} \frac{1}{2}k_\alpha k_{rep} \left(\frac{1}{d(p,p_0)} - \frac{1}{d_0}\right)^2 d_g^2, & 0 \leq d(p, p_0) \leq d_0 \\ & \alpha \in (0, \pi) \\ 0 & , d(p, p_0) > d_0 \end{cases} \quad (30)$$

After introducing the distance factor d_g^2 , the repulsion from the obstacle and the attraction from the target point will both diminish to zero only when the vehicle reaches the target point. This resolves the issue of the target being unreachable when the obsta-

cle is in close proximity to the target point. However, when there are multiple obstacles, they may lead to local minima because their repulsion and gravity may be in the opposite direction. Therefore, the repulsion function of the obstacle is improved, and a repulsion term pointing to the target point is added. The enhanced repulsion function is defined as follows:

$$F_{rep}(d, \alpha) = \begin{cases} F_{rep1} + F_{rep2}, & 0 \leq d(p, p_0) \leq d_0 \cap \alpha \in (0, \pi) \\ 0, & d(p, p_0) > d_0 \end{cases} \quad (31)$$

The F_{rep1} and F_{rep2} expressions are as follows:

$$F_{rep1} = k_{\alpha} k_{rep} \left(\frac{1}{d(p, p_0)} - \frac{1}{d_0} \right) d_g^2 \quad (32)$$

$$F_{rep2} = \frac{k_{\alpha}}{2} k_{rep} \left(\frac{1}{d(p, p_0)} - \frac{1}{d_0} \right) d_g \quad (33)$$

where d_0 represents the range of the obstacle's repulsive potential field, d denotes the distance between the obstacle and the vehicle, d_g represents the distance from the vehicle to the target point, k_{rep} is the repulsive gain coefficient, k_{α} is the repulsive potential field constraint factor, F_{rep1} is the repulsive force of the obstacle pointing to the vehicle, and F_{rep2} is directed from the vehicle to the target point. Figure 14 shows the force analysis of the improved vehicle.

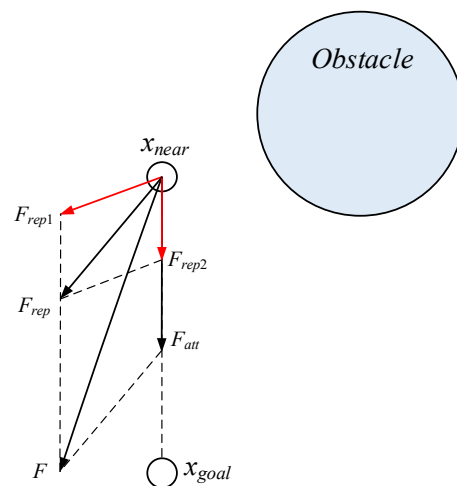


Figure 14. The force of the vehicle after optimizing the repulsion function.

4.4. Virtual Target Point

In the artificial potential field method, the vehicle moves from the high potential field to the low potential field, and the target point is the global minimum point of the potential field, so the vehicle should stop moving at the target point. When there are U-shaped obstacles in the road, the interior is the local minimum region, and the global minimum is the target point. Therefore, the minimum point will not only be at the target point; in this environment, the vehicle driving in the direction of low potential energy will enter the local minimum area of the U-shaped obstacle, which will cause the vehicle to stop or oscillate around this area. The overall potential field featuring a U-shaped obstacle is depicted in Figure 15.

To address this issue, a virtual target point is introduced to steer the vehicle away from obstacles. The potential field is adjusted to improve the passing capacity of the vehicle near the local minimum so as to plan a more reasonable path. When setting the virtual target point, the point is too far from the obstacle, which will lead to the planned path being too long, resulting in an increase in the amount of calculation. The point too close to the obstacle may reduce the safety of the path or even collide with the obstacle. The processing flow for encountering a U-shaped obstacle is shown in Figure 16.

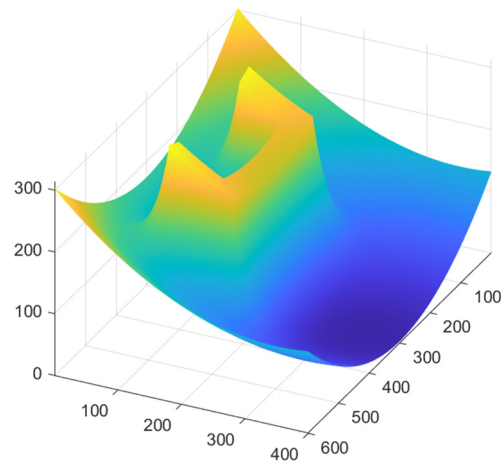


Figure 15. The general potential field near a U-shaped obstacle.

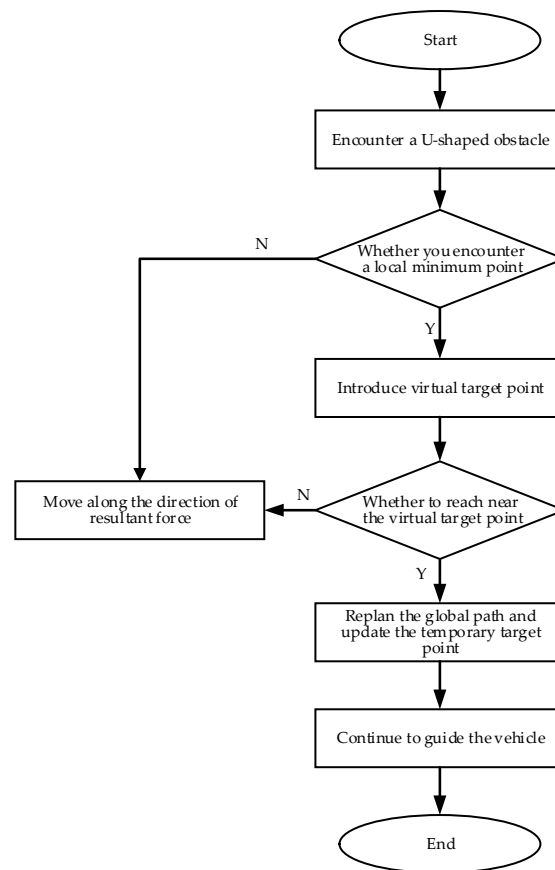


Figure 16. The processing flow of U-shaped obstacles.

Before introducing the virtual target point near the obstacle, first of all, judge whether the vehicle meets the local minimum point or not, and the discrimination condition is as follows:

$$\left| F_{att} + \sum_{j=1}^n F_{rep,j} \right| < \varepsilon \quad (34)$$

$$|x - x_a| < \alpha s_a \quad (35)$$

where j is the number of obstacles, ε is a small positive number, α is a positive number between 0 to 1, x_a is a certain state of the vehicle, and s_a represents the total distance of the vehicle from x_a to the current position x in the process of vehicle movement.

When the determination condition is established, it means that the virtual resultant force on the vehicle is close to 0, and the vehicle has moved for a long distance but its displacement is very small, so it can be considered that the vehicle stops or oscillates around the local minimum [35]. When introducing the virtual target point, the role of the target gravitational potential field is ignored until the vehicle successfully reaches the virtual target point, so as to get rid of the predicament of falling into local optimization in the U-shaped obstacle. The virtual target point is selected where the distance from the obstacle is L_2 longitudinally and L_1 horizontally. The calculation method is as follows:

$$L_1 = \lambda a \quad (36)$$

$$L_2 = \mu b \quad (37)$$

$$L = \sqrt{L_1^2 + L_2^2} \quad (38)$$

where λ and μ are the distance expansion coefficients. According to the danger degree of the obstacle, the safe distance can be adjusted dynamically by changing the value of λ .

Through the above formula, the potential field function can be obtained:

$$U_{vir} = k_{vir} \sqrt{|x - x_0 - L|^2 + |y - y_0|^2} \quad (39)$$

where k_{vir} represents the coefficient representing the potential energy increase of the virtual target point.

Following the identification of the local minimum point, the vehicle becomes influenced by the potential field of the virtual target point. Considering that the target may be unreachable, a circle with a radius of R_v is established as the virtual target area, with the virtual target point as the center. Guided by the global path, the vehicle encounters a U-shaped obstacle and introduces the virtual target point. When the vehicle runs into the area of the virtual target point, it will deactivate the virtual target point and proceed to utilize either the original target point or the updated target point until reaching the global target point. The virtual target point selection method is shown in Figure 17.

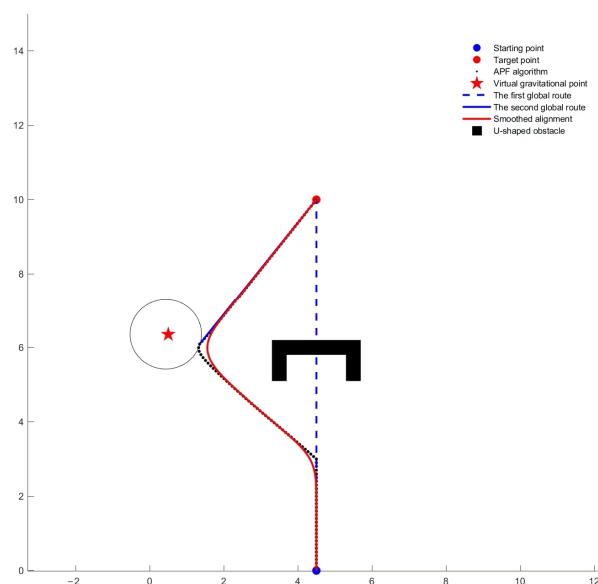


Figure 17. Virtual target point selection method.

On the road, the vehicle is mainly affected by the gravitational potential field, the repulsive potential field, and the road potential field. If the virtual target point is introduced, all the potential fields in the road environment are shown as follows:

$$U_{all} = U_{att} + U_{rep} + U_{road} + U_{vir} \quad (40)$$

4.5. Elliptical Groove Treatment

On structured roads, obstacles typically pose a greater threat in the longitudinal direction than in the lateral direction [36], so this paper uses the obstacle avoidance constraint mentioned in Section 3.1 for the obstacle model. However, when driving on a curved road, the extended elliptical boundary of the obstacle cannot be deflected by the curvature of the road, so it will affect the trajectory of the main vehicle in the lane, resulting in unreasonable planning and even local minimums, as depicted in Figure 18a.

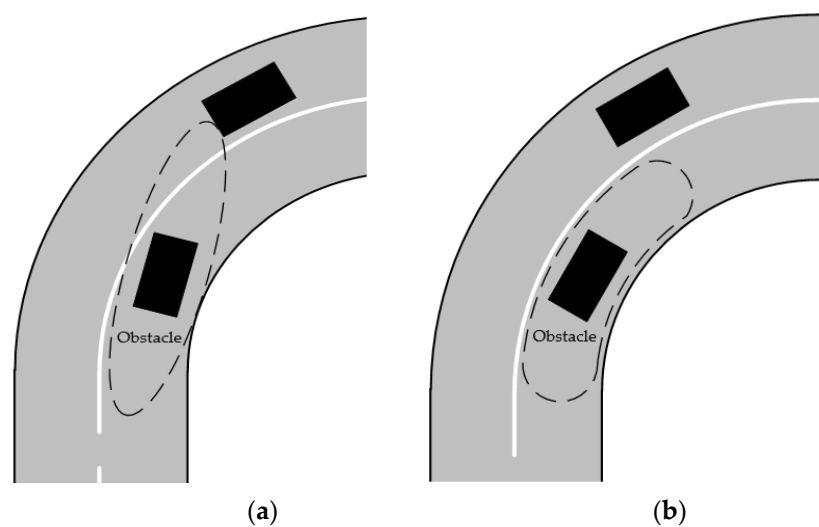


Figure 18. A comparison diagram of elliptical groove treatment: (a) a schematic diagram of the limitations of the repulsive domain of elliptical obstacles; (b) a schematic diagram of an elliptical groove.

To address this issue, the extended ellipse boundary is processed by an elliptical groove by considering the curvature of the road. The center of curvature of the road is taken as the center of the two long boundaries of the elliptical groove, and the bending degree of the two boundaries of the elliptical groove is consistent with the bending degree of the lane line, and then the two long boundaries are closed and connected with two small arcs. Finally, an elliptical groove is formed, as depicted in Figure 18b.

5. Fusion Algorithm

5.1. Analysis of the Limitation of the Algorithm

In addition to the local minimum situation encountered by the artificial potential field method described above, the target is unattainable and there is the problem of facing U-shaped obstacles. There are also algorithmic limitations caused by road conditions.

As the distance between the vehicle and the target point increases, so does the gravitation in the potential field. Affected by the road conditions, when driving on the turning road, A is the starting point and C is the target point. The distance AC between the main car A and the target point C is less than the distance BC between the point B on the way and the target point C. Therefore, the potential field of point B will be larger than that of point A, and it is impossible to travel from the low potential field to the high potential field in the APF potential field, so there will be a local minimum, as illustrated in Figure 19.

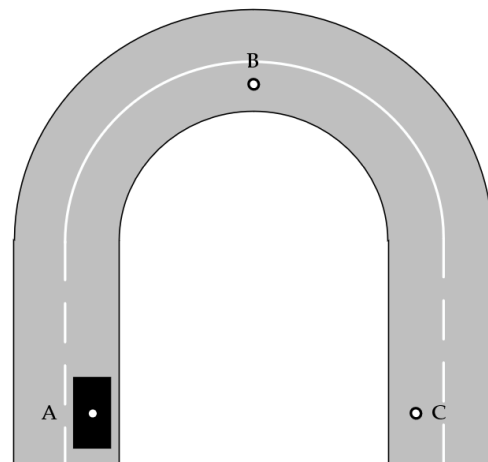


Figure 19. A schematic diagram of the limitations of the turning road.

In the actual driving scene, the environment is usually partially known, but there are often unknown obstacles. Therefore, path planning needs to consider global and local factors. Global path planning algorithms can usually find feasible paths in the whole environment but may not be able to deal with environmental changes and unknown obstacles. The local path planning algorithm can usually quickly respond to environmental changes and obstacle movement, but it only considers local factors, is likely to encounter local optimization, and so on. Therefore, as a local path planning algorithm, the artificial potential field method often falls into the local minimum point in a complex environment and cannot reach the target point smoothly due to insufficient global information. As a global algorithm, the RRT* algorithm is effective in static path planning when the road environment information is known. However, it lacks real-time obstacle avoidance capabilities when confronted with unknown obstacles.

5.2. Algorithm Fusion Strategy

To simultaneously accomplish global path optimization and local avoidance of unknown obstacles, this paper proposes a hybrid path planning approach integrating enhancements to the RRT* algorithm and the artificial potential field method. The RRT* algorithm is improved by using the strategies of probability sampling optimization, adaptive step size, and path post-processing, and the artificial potential field concept is introduced to expedite the discovery of the global path within the road environment. The artificial potential field method is a local obstacle avoidance algorithm for the real-time generation of obstacle avoidance paths that realizes path planning according to the motion state and environment information of the controlled object. Therefore, combining the improved RRT* algorithm and the enhanced artificial potential field approach suggests a fusion algorithm, which can combine the benefits of both global and local path planning algorithms. Before applying the artificial potential field method to dynamic path planning, the improved RRT* algorithm is used for the global path planning of static known environment roads, and the global path node sequence is extracted.

The initial and final points in the node sequence also serve as the starting and ending points for vehicle path planning. Make each node of the node sequence a temporary target point, and when the vehicle is driving along the path, the enhanced artificial potential field method is used to guide the vehicle by taking each node in the node sequence as the target point in turn. Once the vehicle reaches a node, the subsequent node is designated as a temporary target point until the vehicle reaches the global target point. When unknown obstacles are encountered in the driving process, the enhanced artificial potential field method is employed to locally plan the path, thereby accomplishing obstacle avoidance. Upon encountering a local minimum value, the improved RRT* global path is automatically re-planned, or the virtual target point is introduced to update the temporary

target point position and direct the vehicle away from the local minimum region, thus solving the local minimum problem of the artificial potential field method and enhancing the vehicle's obstacle avoidance performance. The final fusion algorithm's workflow is depicted in Figure 20.

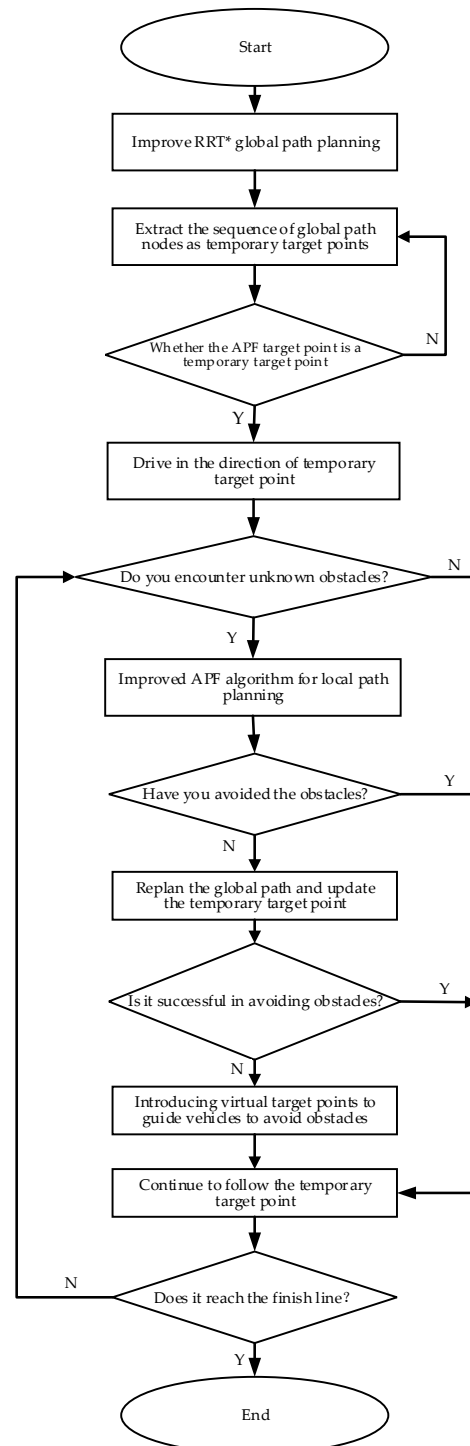


Figure 20. Fusion algorithm flow.

5.3. Path Smoothing Strategy

After the above path optimization, the path produced by the RRT* algorithm exhibits a turning angle, and the turning angle is too large, which is not suitable for vehicles. The

Bezier curve only needs a few control points to generate a more complex, smooth curve [37]. The smooth processing of the path is realized by fitting the points on the path using the n -order Bezier curve. The vehicle can run smoothly.

Given points P_0, P_1, \dots, P_n , the n -th-order Bezier curve is as follows:

$$P(t) = \sum_{i=1}^n P_i B_{iN}(t), t \in [0, 1] \quad (41)$$

where P_i is the vertex coordinate, $B_{iN}(t)$ represents the n -degree Bernstein polynomial, and its basis function is as follows:

$$B_{iN}(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i \quad (42)$$

where $\binom{n}{i} = \frac{n!}{i!(n-i)!}$; Formula (42) represents the n -th order formula of the Bezier curve determined by point $P_0 P_1 \dots P_n$; P_0 , and P_n are the starting point and ending point, respectively; and $P_1 \dots P_{n-1}$ is called the intermediate control point. Each Bezier curve is defined by a set of intermediate control points. In order to make the connection of a multi-segment curve smooth, the end control point and the first end control point of the adjacent segment should be selected so that the tangent direction and curvature of the connection point are continuous and a smooth path can be generated [38].

6. Simulation Analysis

In order to verify whether the improved RRT* algorithm and the fusion algorithm can search the target point and generate the path more effectively at the starting point of the space, the simulation experiments of the improved RRT* algorithm and the fusion algorithm are carried out on the turning road. In order to further verify the performance of the fusion algorithm, the simulation experiments are carried out in a more complex S-bend. The computer is configured with a Windows 11 operating system, an Intel (R) Core (TM) i5-8300H CPU in Dell, Texas, USA, a main frequency of 2.30 GHz, and a running memory of 16.0 GB. The simulation environment is built in the MATLABR2022b environment, and the simulation experiments of the improved RRT* algorithm and fusion algorithm are realized. In order to compare the experimental results, the simulation parameters of each algorithm are unified. The following are the simulation experiment condition settings and parameter selections:

The size of the simulation map in turn is 15×16 . The starting node is (0, 0), and the target node is (9, 0). The size of the simulation map in the S-bend is 26×32 . The starting node is (0, 0), and the target node is (21, 0). The white dotted line and the white solid line are the lanes that can be crossed by vehicles and the lanes that cannot be crossed by vehicles. The grey road area is represented as a safe area, and the black boundary on both sides means that vehicles are not allowed to touch the road boundary. The black rectangular object and the black U-shaped object are unknown obstacles and U-shaped unknown obstacles, respectively, and the red dotted line around the rectangular obstacle is the influence area of the obstacle. The initial step size, the initial target bias probability, the gravitational gain coefficient, the repulsion gain coefficient, the repulsive force range, the maximum number of nodes, and the maximum number of iterations are 0.5, 1.05, 1.15, and 1.25. The maximum number of nodes is 3000, and the maximum number of iterations is 7000. The position of the starting point is represented by a blue solid dot, while the position of the target point is represented by a red solid dot. Considering the randomness of RRT series algorithms, 50 simulation experiments are carried out on each algorithm, and the effectiveness of the improved algorithm is verified by comparing and analyzing the four performance indexes, namely, the average path length, the average number of nodes, the average simulation time, and the average number of iterations.

6.1. Simulation Analysis of Improved RRT* Algorithm

This simulation experiment is divided into Experiments 1 and 2. Experiment 1 is the simulation experiment of the improved RRT* algorithm before path post-processing, and Experiment 2 is the simulation experiment of path post-processing.

6.1.1. Experiment 1

To evaluate the effectiveness of the improved RRT* algorithm proposed in this paper, an analysis was conducted. Simulation experiments were conducted on the conventional RRT algorithm, the RRT* algorithm, and the proposed improved algorithm outlined in this paper. Among them, the traditional improved RRT algorithm, which introduces the concept of potential fields, is used to compare with the improved algorithm in this paper, which is commonly used in the improvement method of the RRT algorithm. The above four RRT algorithms are simulated, respectively, and the average path length, average number of nodes, average simulation time, and average number of iterations are compared and analyzed. To further confirm the efficacy of the enhanced algorithm, the improved algorithm is simulated and compared with the Dijkstra algorithm, the ant colony algorithm, the A* algorithm, and the improved A* algorithm, and the four performance indicators of each algorithm are analyzed. Figure 21 below displays the simulation outcomes of the aforementioned four algorithms. The solid blue line is the search path, and the red solid line is the final search path. Table 3 shows the comparison of the experimental data for the average path length, average number of nodes, average simulation time, and average number of iterations of the four algorithms.

Table 3. Comparative analysis table of performance indexes of RRT algorithm.

Algorithm	Average Path Length	Average Number of Nodes	Average Simulation Time (s)	Average Iterations
RRT algorithm	37.92	81	29.61	6150
RRT* algorithm	34.05	42	20.09	5590
Traditional improved RRT algorithm	33.89	47	2.24	575
This paper improves the algorithm.	32.47	12	1.21	279

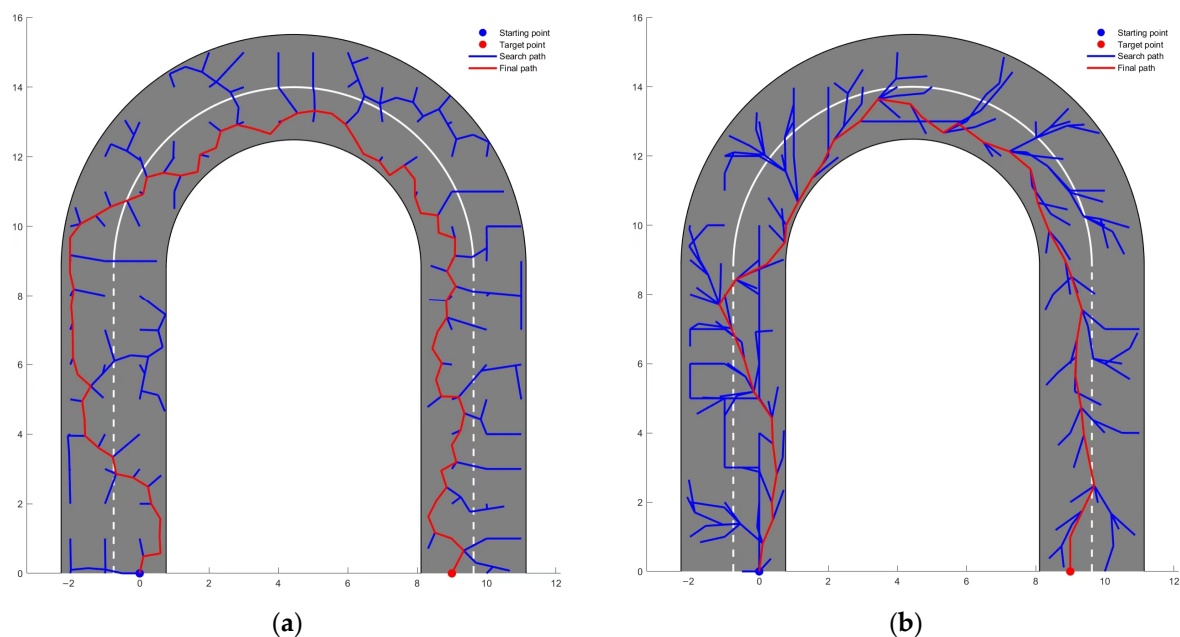


Figure 21. Cont.

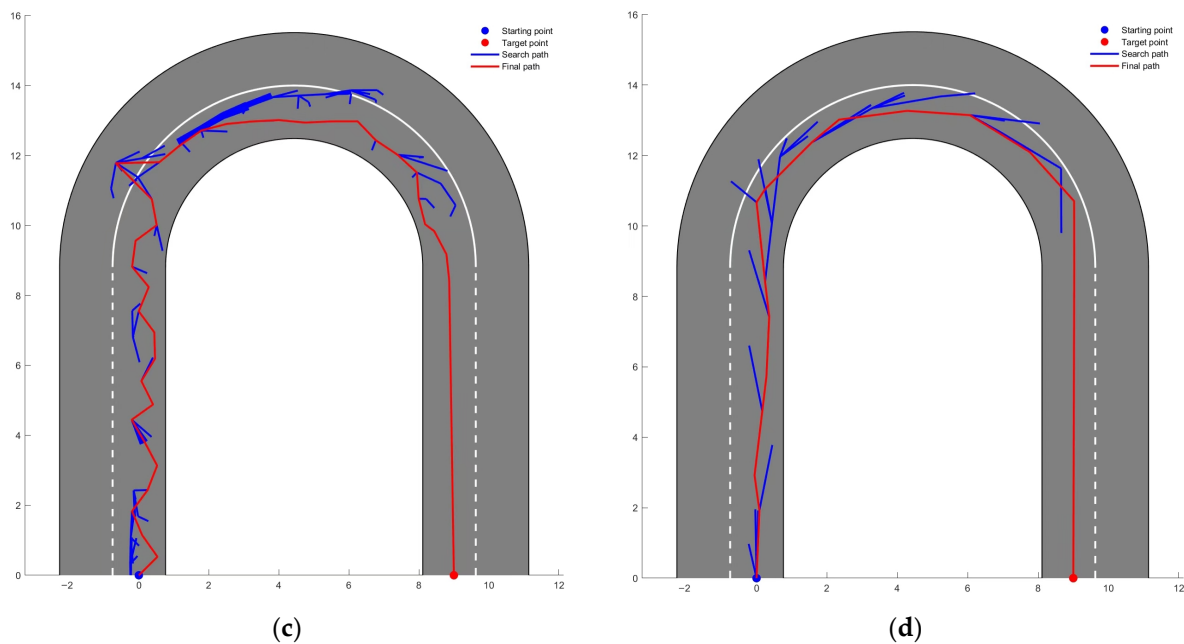


Figure 21. Comparison diagram of improved RRT* algorithm: (a) simulation path of traditional RRT algorithm; (b) RRT* algorithm simulation path; (c) simulation path of traditional improved RRT algorithm; (d) this paper improves RRT* algorithm simulation path.

Compared to the traditional RRT algorithm, the improved RRT* algorithm presented in this paper reduces the average path length by 14.37%, 4.64%, and 4.19% for the RRT* algorithm and the traditional improved RRT algorithm. The reduction in path length is not obvious. This occurs due to the excessive randomness inherent in the RRT algorithm, and there will still be redundant nodes. Moreover, the algorithm enhanced in this paper has not yet undergone the post-processing stage so far in the simulation experiment. However, in terms of the average number of nodes, compared with the RRT algorithm, the RRT* algorithm, and the traditional enhanced RRT algorithm, they were reduced by 85.19%, 71.43%, and 77.47%, respectively. This suggests that the enhanced RRT* algorithm significantly impacts the optimization strategy for redundant path branches, greatly reducing the number of redundant nodes. In terms of average simulation time, the average simulation time is reduced by 95.91%, 93.98%, and 45.98%, respectively, with the RRT algorithm, the RRT* algorithm, and the traditional improved RRT algorithm. At the same time, because the RRT algorithm is a random search algorithm, it also notably diminishes the path's randomness attributed to time cost and can find the global search path more quickly. In terms of the average number of iterations, compared to the RRT algorithm, the RRT* algorithm, and the traditional improved RRT algorithm, the average number of iterations is reduced by 95.46%, 95%, and 51.48%, respectively, and the average number of iterations is significantly reduced, indicating that the path generation speed of the enhanced algorithm is faster, the time cost is lower, and the randomness of the RRT algorithm is reduced. Combined with Figure 21d, it can be seen that while ensuring a safe distance at the road boundary, the improved RRT* algorithm can generate the path more quickly and accurately, and the improved RRT* algorithm significantly reduces redundant nodes in the path, minimizes lateral fluctuations, and enhances path stability. Observation of Figure 21a–c shows that it is obvious that these algorithms have many redundant nodes, have a large invalid search range, and even extend to the opposite lane. The performance indicator comparison of the four algorithms is presented in Table 3.

As depicted in Figure 22, the blue dotted line, purple dashed line, yellow dashed line, black dashed line, and red solid line are the paths planned by the A* algorithm, the improved A* algorithm, the ant colony algorithm, the Dijkstra algorithm, and the improved RRT* algorithm, respectively. It is evident that the improved RRT* algorithm can rapidly

plan the global path compared to the other four algorithms while still maintaining a certain safety margin from the road boundary; it does not encroach on the opposite lane, and the vehicle tends towards the centerline of the lane in which it is situated, and the planned path has more accurate nodes, which play an accurate guiding role for the following fusion algorithm. Among them, the improved A* algorithm takes into account the lane boundary constraints. Although the Dijkstra algorithm and the A* algorithm have a slightly smaller average path length, they do not take into account the impact of the lane boundary on the path. The planned path cannot avoid driving along the lane boundary and colliding with the lane boundary, resulting in a risky path. Due to the influence of the level of pheromone, the path planned by the ant colony algorithm has some limitations, such as a return path, a long planning time, many path turning points, and so on. The enhanced A* algorithm considers the safe distance from the lane boundary and mitigates the risk of driving close to or along the boundary, but does not consider the opposite lane, resulting in the encroachment of the opposite lane; thus, the risk path appears. In comparison to the enhanced RRT* algorithm outlined in this study, the other four algorithms obviously show poor ability to deal with the turning road. The comparison of performance indicators of the five algorithms is depicted in Table 4.

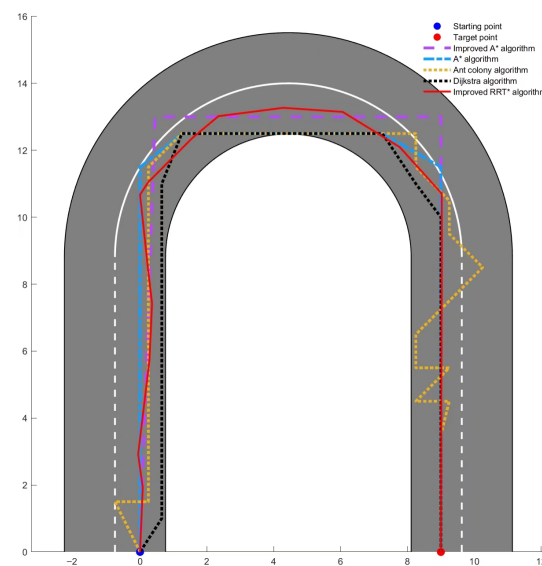


Figure 22. The improved algorithm in this paper and the overall effect simulation path of the four algorithms.

Table 4. Comparative analysis table of performance indexes of five algorithms.

Algorithm	Average Path Length	Average Number of Nodes	Average Simulation Time (s)	Average Iterations
Dijkstra algorithm	31.49	7	1.42	313
A* algorithm	31.65	6	1.37	175
Ant colony algorithm	32.93	16	3.47	-
Improved A* algorithm	31.78	4	1.15	116
This paper improves the algorithm.	32.47	12	1.21	279

6.1.2. Experiment 2

To evaluate the efficacy of the enhanced RRT* algorithm's path post-processing and the effect of the node pruning strategy and node optimization, the enhanced RRT* algorithm and the enhanced RRT* algorithm with path post-processing are both subjected to simulation experiments within identical scenarios to facilitate comparative analysis.

According to Figure 23, it is evident that the average number of nodes after path post-processing is 25% less than that of the original path, and the average path length exhibits a reduction of 18.32% compared to the original path. There are fewer redundant nodes in the optimized path, and the corner of the original path becomes smoother in the bend, which adheres more closely to the vehicle’s kinematic constraints, and the updated node is more accurately close towards the centerline of the lane, which lays a better foundation for the future fusion algorithm. The results indicate that the path generated by the enhanced RRT* algorithm with path post-processing is superior, which is more suitable for guiding vehicles in the future fusion algorithm and directly verifies the effectiveness of path post-processing. The comparative analysis table of path post-processing performance indicators is depicted in Table 5.

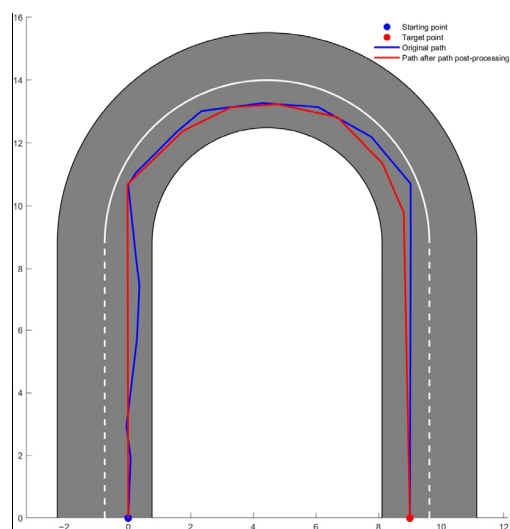


Figure 23. Path post-processing simulation path.

Table 5. Comparative analysis table of performance index of path post-processing.

Algorithm	Average Path Length	Average Number of Nodes
This paper improves the algorithm.	32.47	12
An improved algorithm after node optimization	26.52	9

6.2. Simulation Analysis of Fusion Algorithm

In order to verify the effectiveness of the improved RRT* and improved APF fusion algorithms, the road environments of turning back and S turning are established, respectively, to verify the passing ability of the fusion algorithm for turning back and S turning and to set up sudden, unknown obstacles at different positions on the road. Further analysis of the fusion algorithm ensures that it can pass through the turn while avoiding unknown obstacles smoothly. Finally, a path-smoothing strategy is introduced to the planned path. With or without unknown obstacles and unknown obstacles in straight roads or bends and road scenes with complex obstacles, the passability and obstacle avoidance ability of the fusion algorithm are analyzed and verified. Figure 24 show the simulation results of experiments under different obstacles. Among them, the blue solid line, black point, and red solid line are the paths planned by the improved RRT* algorithm, the fusion algorithm, and the smooth fusion algorithm, respectively.

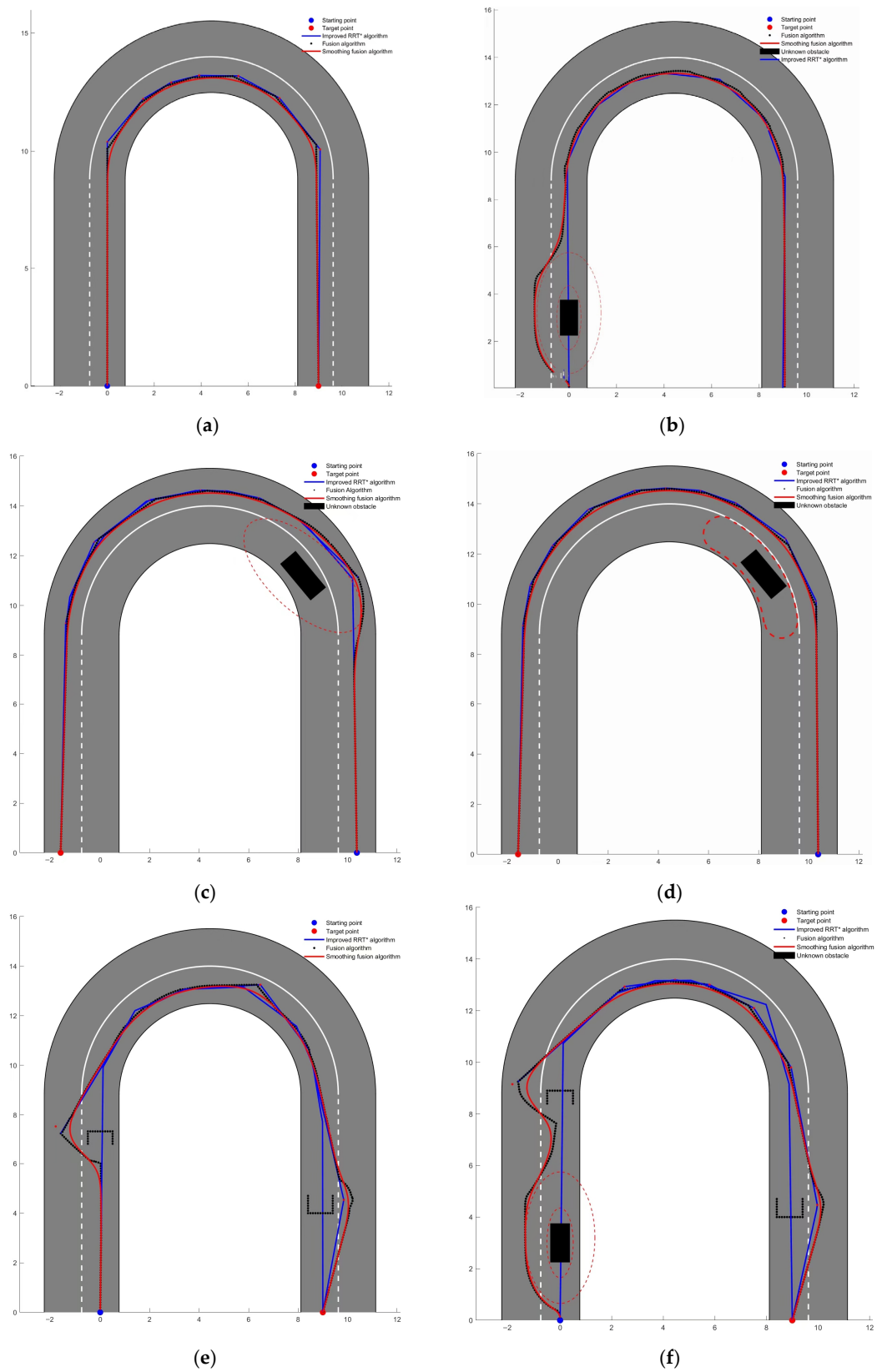


Figure 24. Simulation analysis of fusion algorithm: (a) simulation path of fusion algorithm without obstacle; (b) simulation path of obstacle fusion algorithm on straight road; (c) limited simulation path of safe ellipse of obstacles in bends; (d) simulation path of fusion algorithm for obstacles in bends; (e) simulation path of road fusion algorithm with U-shaped obstacles; (f) simulation path of road fusion algorithm with complex obstacles.

Through the above five groups of simulation experiments, from Figure 24a, it is evident that when utilizing the fusion algorithm, the vehicle can navigate the curved road more smoothly than with the traditional APF algorithm. On the road without unknown obstacles, both the enhanced RRT* algorithm and the fusion algorithm can generate a path from the initial point to the final destination. However, as can be seen from Figure 24b,e, after the unknown obstacles appear on the road, the enhanced RRT* algorithm by itself cannot adequately evade abrupt unknown obstacles, while the fusion algorithm can evade the unknown obstacles and satisfy the ellipse collision constraints. As depicted in Figure 24e, when facing the first U-shaped obstacle, the vehicle is trapped in the local optimal situation. Through the action of the fusion algorithm, a virtual target point is introduced in proximity to the obstacle, and after guiding the vehicle to near the virtual target point, the global path re-planning is realized, and the temporary target point is updated to keep the vehicle going. For the second U-shaped obstacle, it is because the global path is re-planned. By making a detour, the vehicle can be guided to avoid it; additionally, the fusion algorithm demonstrates robustness in obstacle avoidance. As can be seen from Figure 24c,d, before the elliptical slot processing is introduced into the algorithm, the obstacles have some limitations in the curve, which could impact the driving of the main vehicle and make the planned path turn unnecessarily. It even makes the main car fall into the local optimal situation. After the introduction of elliptical slot processing, the path will not be disturbed by obstacles, and the vehicle can navigate the road smoothly. To further validate the efficacy of the fusion algorithm, all obstacles are put on the same turning road. As shown in Figure 24f, the fusion algorithm can avoid unknown obstacles in complex scenes and satisfy collision constraints. The two U-shaped obstacles are also reasonably avoided. When reaching near the virtual target point, it can re-plan a global path and successfully guide to the target point. Simultaneously, the path planned by the above experiment is smoothed to meet the requirements of vehicle tracking.

To further corroborate the effectiveness and robustness of the fusion algorithm, the road scene is changed from a single U-turn to a complex S-bend. The simulation results from Figure 25 show that no matter the barrier-free S-bend road, the simple obstacle S-bend road, or the complex obstacle S-bend road, the fusion algorithm can still safely and reasonably avoid random obstacles and reach the target point. In Figure 25c, when the last obstacle is planned by the fusion algorithm, the improved RRT* is re-planned twice. In the first re-planning, the path node falls within the scope of the collision constraint, so it is eliminated. After the second re-planning, the node is outside the constraint range, and the vehicle is successfully guided to avoid the obstacle. At the same time, the above S-bend road simulation experiments ensure the smoothness of the planned path to fulfil the vehicle's tracking requirements. All of the aforementioned experimental results underscore the indispensability and superiority of the fusion algorithm.

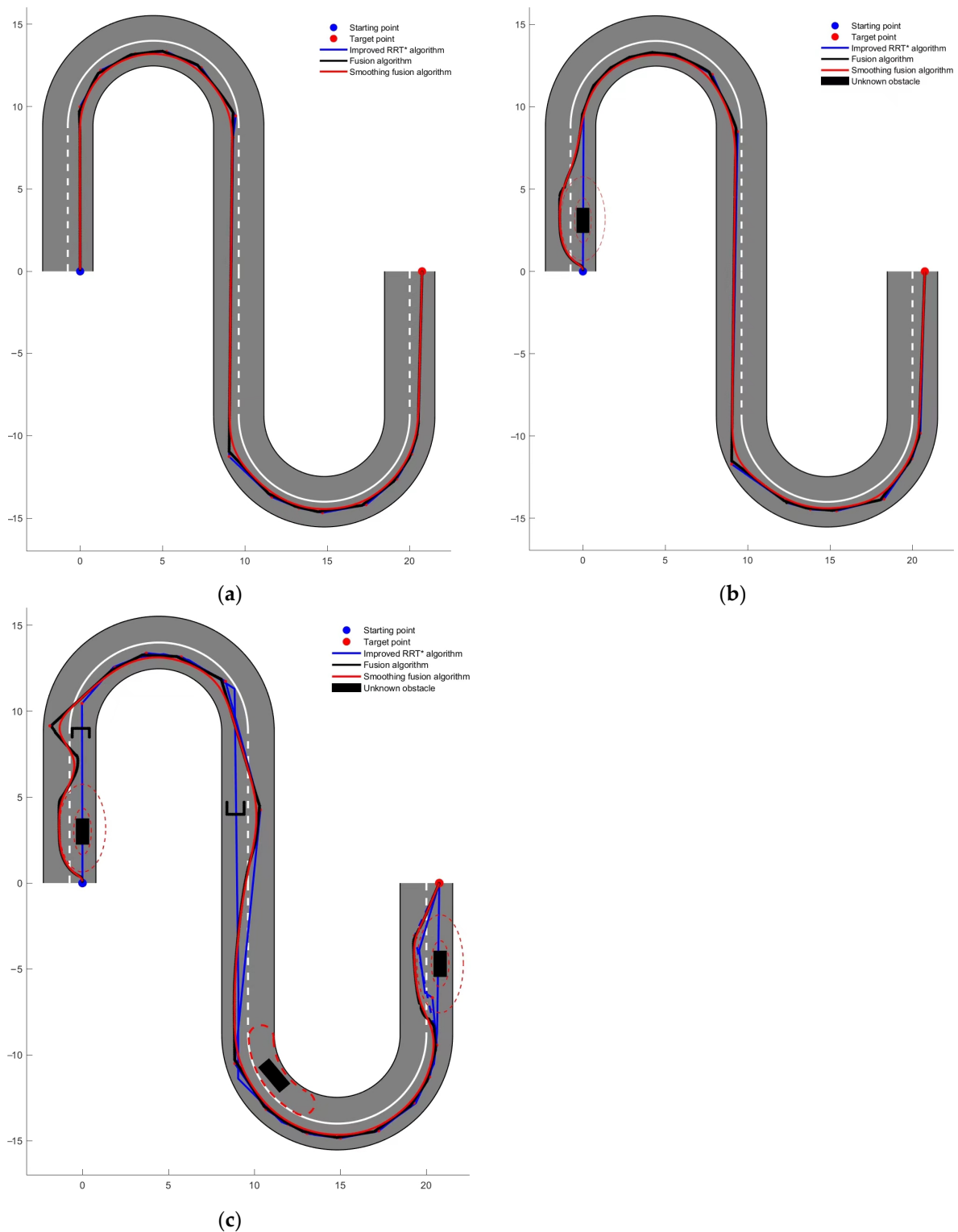


Figure 25. Simulation analysis of fusion algorithm: (a) simulation path of S-bend obstacle-free scene fusion algorithm; (b) simulation path of S-bend simple obstacle scene fusion algorithm; (c) simulation path of S-bend complex obstacle scene fusion algorithm.

7. Conclusions

For the purpose of solving the path planning problem of highlighting unknown obstacles on the structured road, this paper takes the improved RRT* algorithm and the improved artificial potential field method as the main body of research, examines the path

planning algorithm, improves the global path search efficiency and path quality of the RRT* algorithm, and solves the limitations of the artificial potential field method of local path planning. Combined with the two algorithms, a fusion algorithm is devised for global dynamic scene path planning.

- (1) Targeting the deficiencies of the RRT* algorithm, such as strong randomness, slow convergence speed, poor path feasibility, and poor ability to deal with corners, the RRT* algorithm is improved, and the concepts of artificial potential field and probabilistic sampling optimization are introduced to make RRT* node sampling more purposeful, probabilistic sampling applicability stronger, and sampling efficiency better. Considering the constraint imposed by the fixed step size, the adaptive step size is designed according to the road curvature to solve the problem that oscillation may occur near the target point, improve the adaptability to each road scene, and achieve rapid convergence towards the target point accurately. The path planned by the improved RRT* is post-processed to minimize the number of redundant nodes along the path and optimize the global path quality.
- (2) In view of the problems that the artificial potential field method is prone to local optimization, the target is unreachable, and it is not suitable for the global scene, an enhanced artificial potential field method is introduced, which adds obstacle avoidance constraints to obstacles and formulates a road boundary repulsion potential field to delineate the risk boundaries within the road space. The distance factor is added to the repulsion function to solve the problem of target unreachability. In the face of U-shaped obstacles, virtual gravity points are introduced to solve the local minimum problem and improve the performance of obstacle avoidance. For the case where the obstacle is located in the bend, the safety ellipse of the obstacle is treated with an elliptical slot to reduce unnecessary steering on the planned path.
- (3) Aiming at the emerging unknown obstacles, complex obstacles, and other road scenes, a fusion algorithm of improved RRT* and an improved artificial potential field method is proposed. The enhanced RRT* algorithm is employed for generating the global path, and the nodes of the global path are extracted to serve as temporary target points for the artificial potential field method, which guides the vehicle to drive, reduces the occurrence of the local minimum, uses the artificial potential field method to avoid the local path when encountered with unknown obstacles, and carries on the path smoothing processing to the planned path to satisfy the vehicle's driving. The simulation results in different scenarios demonstrate that the fusion algorithm can successfully plan a smoother and more feasible path and verify its effectiveness and robustness.

Currently, this study has its limitations. In our future work, we will consider adding the fusion algorithm to decision making in more complex road environments, such as crossroads, and analyze its path-planning effect.

Author Contributions: Conceptualization, X.L. and G.L.; methodology, X.L. and Z.B.; software, Z.B. and X.L.; validation, X.L., G.L., and Z.B.; formal analysis, X.L.; investigation, Z.B.; resources, G.L. and Z.B.; data curation, X.L. and Z.B.; writing—original draft preparation, X.L.; writing—review and editing, X.L. and G.L.; visualization, X.L. and Z.B.; supervision, G.L.; project administration, G.L.; funding acquisition, G.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by key research projects of China Liaoning Provincial Department of Education (JYTZD2023081), China Liaoning Provincial Natural Fund Grant Program Project (2022-MS-376) and Key Projects of China National Natural Science Foundation Joint Fund (U22A2043).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Achiri, A.T.; Mbue, I.N.; Merlin, A.; Gerard, A. Automobile Crash Investigation Based on Vehicle System Related Causes: Systematic Literature Review. *World J. Eng. Technol.* **2022**, *10*, 139–157. [[CrossRef](#)]
2. Gaba, S.; Pawar, S.; Chinta, P.P.; Sweta, S. A computational model of deep learning in self-driving car. *AIP Conf. Proc.* **2023**, *2755*, 020019. [[CrossRef](#)]
3. Lin, S.-L.; Li, X.-Q.; Wu, J.-Y.; Lin, B.-C. Research on overtaking path planning of autonomous vehicles. In Proceedings of the 2021 IEEE International Future Energy Electronics Conference (IFEEEC), Taipei, Taiwan, 16–19 November 2021; pp. 1–4. [[CrossRef](#)]
4. Reda, M.; Onsy, A.; Haikal, A.Y.; Ghanbari, A. Path planning algorithms in the autonomous driving system: A comprehensive review. *Robot. Auton. Syst.* **2024**, *174*, 104630. [[CrossRef](#)]
5. Li, X. Path planning of intelligent mobile robot based on Dijkstra algorithm. *J. Phys. Conf. Ser.* **2021**, *2083*, 042034. [[CrossRef](#)]
6. He, Z.; Liu, C.; Chu, X.; Negenborn, R.R.; Wu, Q. Dynamic anti-collision A-star algorithm for multi-ship encounter situations. *Appl. Ocean. Res.* **2022**, *118*, 102995. [[CrossRef](#)]
7. Li, Q.; Xu, Y.; Bu, S.; Yang, J. Smart Vehicle Path Planning Based on Modified PRM Algorithm. *Sensors* **2022**, *22*, 6581. [[CrossRef](#)] [[PubMed](#)]
8. Huang, G.; Ma, Q. Research on Path Planning Algorithm of Autonomous Vehicles Based on Improved RRT Algorithm. *Int. J. Intell. Transp. Syst. Res.* **2021**, *20*, 170–180. [[CrossRef](#)]
9. Karaman, S.; Frazzoli, E. Sampling-based algorithms for optimal motion planning. *Int. J. Robot. Res.* **2011**, *30*, 846–894. [[CrossRef](#)]
10. Cong, J.; Hu, J.; Wang, Y.; He, Z.; Han, L.; Su, M. FF-RRT*: A sampling-improved path planning algorithm for mobile robots against concave cavity obstacle. *Complex Intell. Syst.* **2023**, *9*, 7249–7267. [[CrossRef](#)]
11. Lin, Y.; Zhang, L. An improved Quick Informed-RRT* algorithm based on hybrid bidirectional search and adaptive adjustment strategies. *Intell. Serv. Robot.* **2024**. [[CrossRef](#)]
12. Gu, X.; Han, M.; Zhang, W.; Xue, G.; Zhang, G.; Han, Y. Intelligent vehicle path planning based on improved artificial potential field algorithm. In Proceedings of the 2019 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS), Shenzhen, China, 9–11 May 2019; pp. 104–109. [[CrossRef](#)]
13. Li, Y.; Yang, W.; Zhang, X.; Kang, X.; Li, M. Research on Automatic Driving Trajectory Planning and Tracking Control Based on Improvement of the Artificial Potential Field Method. *Sustainability* **2022**, *14*, 12131. [[CrossRef](#)]
14. Xu, X.; Hu, Y.; Zhai, J.M.; Li, L.Z.; Guo, P.S. A novel non-collision trajectory planning algorithm based on velocity potential field for robotic manipulator. *Int. J. Adv. Robot. Syst.* **2018**, *15*, 1729881418787075. [[CrossRef](#)]
15. Zhai, S.; Pei, Y. The Dynamic Path Planning of Autonomous Vehicles on Icy and Snowy Roads Based on an Improved Artificial Potential Field. *Sustainability* **2023**, *15*, 15377. [[CrossRef](#)]
16. Huang, S. Path planning based on mixed algorithm of RRT and artificial potential field method. In Proceedings of the 2021 4th International Conference on Intelligent Robotics and Control Engineering (IRCE), Lanzhou, China, 18–20 September 2021; pp. 149–155. [[CrossRef](#)]
17. Zhang, W.; Wang, N.; Wu, W. A hybrid path planning algorithm considering AUV dynamic constraints based on improved A* algorithm and APF algorithm. *Ocean. Eng.* **2023**, *285*, 115333. [[CrossRef](#)]
18. Wu, H.; Zhang, Y.; Huang, L.; Zhang, J.; Luan, Z.; Zhao, W.; Chen, F. Research on vehicle obstacle avoidance path planning based on APF-PSO. *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.* **2022**, *237*, 1391–1405. [[CrossRef](#)]
19. Dasiah, N.B.J.a.p.a. Modified RRT* for Path Planning in Autonomous Driving. *arXiv* **2024**, arXiv:arXiv:2402.12129. [[CrossRef](#)]
20. Nguyen, K.; Dang, V.T.; Pham, D.D.; Dao, P.N. Formation control scheme with reinforcement learning strategy for a group of multiple surface vehicles. *Int. J. Robust Nonlinear Control.* **2023**, *34*, 2252–2279. [[CrossRef](#)]
21. Shi, Y.; Dong, X.; Hua, Y.; Yu, J.; Ren, Z. Distributed output formation tracking control of heterogeneous multi-agent systems using reinforcement learning. *ISA Trans.* **2023**, *138*, 318–328. [[CrossRef](#)] [[PubMed](#)]
22. Wei, K.; Ren, B. A Method on Dynamic Path Planning for Robotic Manipulator Autonomous Obstacle Avoidance Based on an Improved RRT Algorithm. *Sensors* **2018**, *18*, 571. [[CrossRef](#)]
23. Xin, P.; Wang, X.; Liu, X.; Wang, Y.; Zhai, Z.; Ma, X. Improved Bidirectional RRT* Algorithm for Robot Path Planning. *Sensors* **2023**, *23*, 1041. [[CrossRef](#)]
24. Dong, M.; Chen, T.; Yang, J. Simulation Research on path Planning of Unmanned vehicle based on improved RRT algorithm. *Comput. Simul.* **2019**, *36*, 96–100.
25. Zhuge, C.; Wang, Q.; Liu, J.; Yao, L. An Improved Q-RRT* Algorithm Based on Virtual Light. *Comput. Syst. Sci. Eng.* **2021**, *39*, 107–119. [[CrossRef](#)]
26. Li, W.; Jin, J. Optimal path Convergence method based on artificial potential Field method and heuristic sampling. *Comput. Appl.* **2021**, *41*, 2912. [[CrossRef](#)]
27. Liu, X.; Cao, L. Research on path Planning of manipulator based on improved RRT* algorithm. *J. Sichuan Univ. Sci. Eng. (Nat. Sci. Ed.)* **2024**, *37*, 61–70. [[CrossRef](#)]
28. Zhang, Y.; Gao, F.; Zhao, F. Research on Path Planning and Tracking Control of Autonomous Vehicles Based on Improved RRT* and PSO-LQR. *Processes* **2023**, *11*, 1841. [[CrossRef](#)]

29. Wang, J.; Li, B.; Meng, M.Q.H. Kinematic Constrained Bi-directional RRT with Efficient Branch Pruning for robot path planning. *Expert Syst. Appl.* **2021**, *170*, 114541. [[CrossRef](#)]
30. Guo, Y.; Liu, H.; Fan, X.; Lyu, W.; Chen, H. Research Progress of Path Planning Methods for Autonomous Underwater Vehicle. *Math. Probl. Eng.* **2021**, *2021*, 8847863. [[CrossRef](#)]
31. Liu, H. Research on Path Planning of RRT Algorithm Based on APF. Master's Thesis, Jiangsu University, Zhenjiang, China, 2022. [[CrossRef](#)]
32. Wang, W.; Li, G.; Liu, S.; Yang, Q. Trajectory Planning of a Semi-Trailer Train Based on Constrained Iterative LQR. *Appl. Sci.* **2023**, *13*, 10614. [[CrossRef](#)]
33. Hongyu, H.; Chi, Z.; Yuhuan, S.; Bin, Z.; Fei, G.J.I.-P. An improved artificial potential field model considering vehicle velocity for autonomous driving. *IFAC-PapersOnLine* **2018**, *51*, 863–867. [[CrossRef](#)]
34. Ruan, S. *Research on Path Planning of Mobile Robot Based on Improved RRT* and Artificial Potential Field Method*; Hangzhou University of Electronic Science and Technology: Hangzhou, China, 2023. [[CrossRef](#)]
35. Xie, C.; Tao, T.; Li, J. Research on path Planning based on improved artificial potential Field method. *J. Jilin Univ. (Inf. Sci. Ed.)* **2023**, *41*, 998–1006. [[CrossRef](#)]
36. Li, Y.; Jin, R.; Xu, X.; Qian, Y.; Wang, H.; Xu, S.; Wang, Z. A Mobile Robot Path Planning Algorithm Based on Improved A* Algorithm and Dynamic Window Approach. *IEEE Access* **2022**, *10*, 57736–57747. [[CrossRef](#)]
37. Lai, R.; Wu, Z.; Liu, X.; Zeng, N. Fusion Algorithm of the Improved A* Algorithm and Segmented Bézier Curves for the Path Planning of Mobile Robots. *Sustainability* **2023**, *15*, 2483. [[CrossRef](#)]
38. Li, X.; Zheng, H.; Wang, J.; Xia, Y.; Song, H. A Global Path Planning Method for Unmanned Ground Vehicles in Off-Road Scenarios Based on Terrain Data. *Soc. Sci. Res. Netw.* **2024**. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.